



TUGAS AKHIR - TE 141599

**RANCANG BANGUN SISTEM MONITORING KELAS
TERINTEGRASI BERBASIS RASPBERRY PI DAN
WEBCAM SEBAGAI PENDETEKSI KECURANGAN SAAT
UJIAN**

Andri Yasmin
NRP 2211100194

Dosen Pembimbing
Suwito, ST., MT.
Ronny Mardiyanto, ST., MT., Ph.D

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - TE 141599

**DEVELOPMENT OF RASPBERRY PI AND WEBCAM
BASED INTEGRATED CLASS MONITORING SYSTEM AS
FRAUD DETECTOR IN EXAMINATION**

Andri Yasmin
NRP 2211100194

Supervisor
Suwito, ST., MT.
Ronny Mardiyanto, ST., MT., Ph.D

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2015

**RANCANG BANGUN SISTEM MONITORING
KELAS TERINTEGRASI BERBASIS
RASPBERRY PI DAN *WEBCAM* SEBAGAI
PENDETEKSI KECURANGAN SAAT UJIAN**

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada

Bidang Studi Elektronika
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Menyetujui :

Dosen Pembimbing I,



Suwito, ST., MT.

NIP. 198101052005011004

Dosen Pembimbing II,



Ronny Mardiyanto, ST., MT., Ph.D.

NIP. 198101182003121003



RANCANG BANGUN SISTEM MONITORING KELAS TERINTEGRASI BERBASIS RASPBERRY PI DAN *WEBCAM* SEBAGAI PENDETEKSI KECURANGAN SAAT UJIAN

Nama : Andri Yasmin
Pembimbing I : Suwito, ST., MT.
Pembimbing II : Ronny Mardiyanto, ST., MT., Ph.D

ABSTRAK

Kecurangan dalam dunia pendidikan bukanlah hal yang aneh. Kecurangan ini memiliki banyak bentuk, akan tetapi kecurangan yang paling umum ialah kecurangan saat ujian. Hal ini sangat bertentangan dengan salah satu fungsi sekolah, yaitu membentuk moral siswa-siswi.

Terdapat banyak cara untuk menanggulangi kecurangan ini, mulai dari penempatan guru, pemberian jarak antar murid, hingga pemberian soal yang berbeda-beda. Dengan mengintegrasikan beberapa Raspberry Pi dengan kamera dan dihubungkan dengan *server* via wifi, maka akan dibuat sistem monitoring kelas terintegrasi yang dapat dilihat langsung dari *server* sehingga dapat memonitor beberapa kelas sekaligus. Sistem ini juga memiliki beberapa fitur yang membuatnya lebih unggul jika dibandingkan dengan CCTV pada umumnya, yaitu sistem pengiriman suara ke berbagai *channel* dan sistem pendeteksi gerakan yang dapat memonitor apakah ada gerakan mencurigakan atau tidak.

Dari hasil pengujian yang dilakukan, pengiriman gambar terjadi secara langsung, akan tetapi *frame per second* maksimal berbeda antara Raspberry Pi B+ dan Raspberry Pi 2. Raspberry Pi 2 dapat mencapai 4 *frame per second* tanpa pengiriman, sedangkan Raspberry Pi B+ dapat mencapai 3 *frame per second* tanpa pengiriman. Jika terjadi kecurangan, maka akan terdapat gerakan yang banyak dan cepat yang kemudian akan terdeteksi. Jarak efektif pendeteksi gerakan ialah 2 hingga 10 meter. Presentase gerakan bergantung pada batasan yang diatur. Apabila batasan diatur 4000 dengan jarak 5 meter, maka gerakan akan terdeteksi apabila objek bergerak sebanyak 25%.

Kata kunci : Raspberry Pi, Sistem Monitoring Terintegrasi, *Background Subtraction*

Halaman ini sengaja dikosongkan

DEVELOPMENT OF RASPBERRY PI AND WEBCAM BASED INTEGRATED CLASS MONITORING SYSTEM AS FRAUD DETECTOR IN EXAMINATION

Name : ***Andri Yasmin***
1st Supervisor : ***Suwito, ST., MT.***
2nd Supervisor : ***Ronny Mardiyanto, ST., MT., Ph.D***

ABSTRACT

Fraud, in education world, is very common. This fraud has many forms, but the most common one is cheating in examination. Cheating has many forms, ranging from working with a friend, using notes, to use existing technology, such as mobile phones. This is contradictory to one of the school's function, which is shaping the moral of students.

There are many ways to handle cheating, ranging from placing teacher in each class, giving distance between students, to giving different sheet. By integrating some of Raspberry Pis with cameras and connect them into a server via wifi, it is possible to create an integrated class monitoring system that can be viewed directly from the server so that it can monitor several classes at once. This system has a few features that differ it from ordinary CCTV, which is audio transfer that can send audio into two different channels, and motion detector that can detect suspicious movement.

From the results of tests performed, sending images occur directly, but frames per second maximum of the difference between the Raspberry Pi B + and Raspberry Pi 2. Raspberry Pi 2 can reach 4 frames per second without delivery, while the Raspberry Pi B + can reach 3 frames per second without delivery. Motion can only be detected if there are a lot of fast movement. Motion detection effective range is 2 to 10 meters. Percentage movement depends on the threshold. If the threshold is set 4000 at a distance of 5 meters, then the movement will be detected when a moving object as much as 25%.

Keywords : Raspberry pi, Integrated Monitoring System , Background Subtraction

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur penulis haturkan kepada Tuhan Yang Maha Esa, karena berkatnya penulis berhasil menyelesaikan buku tugas akhir yang berjudul “Rancang Bangun Sistem Monitoring Kelas Terintegrasi Berbasis Raspberry Pi dan *Webcam* sebagai Pendeteksi Kecurangan Saat Ujian”.

Dalam penyusunan buku maupun penyelesaian tugas akhir ini penulis mendapatkan banyak halangan, namun dapat penulis atasi karena bantuan banyak pihak. Ucapan terima kasih penulis terutama penulis tujukan kepada:

- Ayah, Ibu, kakak-kakak, serta seluruh keluarga yang telah memberikan dukungan baik moral maupun material.
- Bapak Suwito selaku dosen pembimbing pertama, yang sangat banyak membantu, baik dalam moral, perancangan, maupun pemberian saran.
- Bapak Ronny Mardiyanto selaku dosen pembimbing kedua, yang juga telah banyak membantu penulis dalam pengembangan dan pemberian saran sehingga tugas akhir ini dapat terwujud.
- Bapak Tasripan selaku Koordinator Bidang Studi Elektronika dan dosen wali yang memberikan begitu banyak motivasi dan inspirasi.
- Bapak Tri Arief Sardjono selaku Ketua Jurusan Teknik Elektro ITS Surabaya.
- Seluruh dosen bidang studi elektronika dan seluruh dosen Teknik Elektro ITS.
- Teman-teman bidang studi elektronika dan teman-teman bidang studi lainnya.

Penulis sepenuhnya menyadari bahwa tugas akhir ini jauh dari sempurna. Saran dan kritik yang membangun senantiasa penulis terima guna pengembangan lebih lanjut. Penulis berharap agar Tugas Akhir ini dapat memberikan manfaat sebesar-besarnya bagi semua pihak.

Surabaya, Juli 2015

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

Halaman

HALAMAN JUDUL	
HALAMAN PENGESAHAN	
ABSTRAK	i
<i>ABSTRACT</i>	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
 BAB 1 PENDAHULUAN	 1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah.....	2
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	4
1.7 Relevansi	4
 BAB 2 DASAR TEORI	 5
2.1 Raspberry Pi	5
2.2 Webcam	7
2.3 Speaker	8
2.4 Router	9
2.5 WLAN	10
2.6 TCP/IP	10
2.7 OpenCV	11
2.8 Python	12
2.9 Web Server Apache.....	13
2.10 Database MySQL	14
2.11 PHP	15
 BAB 3 PERANCANGAN SISTEM	 17
3.1 Diagram Blok	17
3.2 Router dan Wifi Adapter	18
3.2.1 Router TP-Link TL-MR3220	18
3.2.2 Wifi Adapter TP-Link WN725N	20

3.3 Perancangan <i>Database</i>	21
3.4 Perancangan Pengiriman Gambar	22
3.5 Perancangan Penampilan Gambar	23
3.6 Perancangan Pengiriman Suara	26
3.7 Perancangan Pendeteksi Gerakan.....	27
3.8 Perancangan <i>Webpage</i>	29
 BAB 4 PENGUJIAN DAN ANALISIS	 33
4.1 Pengujian Pengiriman Gambar	33
4.2 Pengujian Pengiriman Gambar	34
4.3 Pengujian Pengiriman Suara	35
4.4 Pengujian Pendeteksi Gerakan.....	36
4.5 Pengujian Sistem.....	38
 BAB 5 KESIMPULAN DAN SARAN	 41
5.1 Kesimpulan	41
5.2 Saran	41
 DAFTAR PUSTAKA	 43
LAMPIRAN	45
BIODATA PENULIS	51

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Raspberry Pi B+ dan 2	5
Gambar 2.2 Kamera Web	7
Gambar 2.3 Pengeras Suara Dinamis	8
Gambar 2.4 Sistem <i>Router</i> Statis	9
Gambar 2.5 Perbandingan Model Referensi DARPA dan OSI	11
Gambar 2.6 Logo OpenCV	12
Gambar 2.7 Logo Python	12
Gambar 2.8 Logo Web Server Apache	13
Gambar 2.9 Logo MySQL	14
Gambar 3.1 Diagram Blok	17
Gambar 3.2 TP-Link TL-MR3220	18
Gambar 3.3 TP-Link WN725N	20
Gambar 3.4 <i>Database</i> “raspberrypi”	22
Gambar 3.5 Tabel “upload”	22
Gambar 3.6 Flowchart Pengiriman Gambar	23
Gambar 3.7 Flowchart <i>Download</i> Gambar	24
Gambar 3.8 Flowchart Penampilan Gambar	25
Gambar 3.9 <i>Script Download</i> Gambar	26
Gambar 3.10 Pengaturan <i>Pulseaudio</i>	27
Gambar 3.11 Flowchart Pendeteksi Gerakan	28
Gambar 3.12 Desain <i>Webpage</i>	29
Gambar 3.13 <i>Script</i> Penampilan Gambar	30
Gambar 3.14 <i>Script</i> Penampilan Status	30
Gambar 4.1 <i>Database</i> Pengiriman Gambar	33
Gambar 4.2 Folder Penimpanan Gambar	34
Gambar 4.3 Hasil Penampilan Gambar	35
Gambar 4.4 Tampilan Pengiriman Suara	35
Gambar 4.5 Pengujian Pendeteksi Gerakan	37
Gambar 4.6 Grafik Hubungan Intensitas Cahaya dan Bit Putih	37
Gambar 4.7 <i>Database</i>	38
Gambar 4.8 Sistem Berjalan	39
Gambar 4.9 Simulasi Ujian	40

Halaman ini sengaja dikosongkan#

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi komunikasi telah berkembang pesat. Peralatan komunikasi seperti telepon genggam bukanlah peralatan yang mewah. Fungsi telepon genggam menjadi semakin beragam karena mengikuti kebutuhan pasar. Fitur-fitur yang lazim ditemui pada telepon genggam antara lain kamera, pengiriman pesan singkat (*chatting*), *video call*, dan koneksi internet. Hal ini membuat pertukaran informasi menjadi semakin cepat dan efisien.

Perkembangan teknologi komunikasi ini tentu sangat menguntungkan. Pencarian informasi dapat dengan mudah dilakukan melalui internet. Komunikasi juga dapat dilakukan dengan mudah dan cepat dengan jarak yang sangat jauh. Bahkan hal-hal yang penting seperti rapat dan perkuliahan dapat dilakukan tanpa harus tatap muka secara langsung. Sayangnya, perkembangan teknologi ini kerap disalahgunakan. Salah satu bentuk penyalahgunaan teknologi adalah tindak kecurangan saat ujian.

Kecurangan dalam ujian bukanlah hal yang aneh dan telah banyak dilakukan. Kecurangan-kecurangan tersebut memiliki banyak bentuk, diantaranya kerja sama dengan teman, membuka catatan ditengah ujian, hingga penggunaan alat komunikasi seperti telepon genggam. Bahkan siswa-siswipun kadang-kadang lebih mempersiapkan strategi kecurangan yang akan dilakukan. Tindak kecurangan yang sebelumnya tabu pun perlahan-lahan mulai dimaklumi. Hal ini sungguh ironis mengingat salah satu peran sekolah adalah membangun moral siswa-siswinya.

Cara-cara untuk menanggulangi tindak kecurangan itupun mulai bermunculan, seperti pengetatan pengawasan, pengumpulan benda-benda elektronik hingga pemberian soal yang berbeda-beda. Salah satu cara yang mulai diterapkan yaitu penggunaan CCTV.

CCTV sendiri merupakan teknologi yang kerap digunakan sebagai sistem keamanan. Akan tetapi, CCTV juga dapat dikembangkan menjadi sistem monitoring atau sistem pengawasan. Salah satu pengembangannya adalah penambahan sistem pengiriman suara yang dapat digunakan sebagai pengiriman peringatan. Pengembangan sistem

ini juga harus terjangkau mengingat hanya sekolah elit yang memiliki sistem ini dikarenakan mahalnnya sistem monitoring ini.

Penggunaan sistem ini dapat menangkap siswa-siswi yang berbuat curang dan menahan minat mereka untuk melakukannya lagi. Dengan perancangan sistem ini diharapkan dapat mengurangi tindak kecurangan dan mengembalikan masa-masa dimana kecurangan dalam ujian merupakan hal yang salah, bukan hal yang dapat dimaklumi, sehingga sedikit demi sedikit dapat mengembalikan nilai moral yang dulu ada.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, dapat dirumuskan beberapa masalah, antara lain :

1. Cara Raspberry Pi dapat memproses gambar dan mengirimkan kepada *server*.
2. Cara *server* mengirimkan suara menuju Raspberry Pi.
3. Cara Raspberry Pi mendeteksi gerakan.
4. Pengintegrasian perangkat keras, Raspberry Pi, dan *server*.

1.3 Tujuan Penelitian

Penelitian tugas akhir ini bertujuan sebagai berikut:

1. Membangun sistem monitoring kelas berbasis Raspberry Pi yang terjangkau.
2. Mewujudkan sistem monitoring yang juga dapat mengirimkan suara dan mendeteksi gerakan.
3. Membangun sistem yang dapat mencegah kecurangan dalam kelas.

1.4 Batasan Masalah

Batasan masalah dari tugas akhir ini adalah sebagai berikut:

1. Performa Raspberry Pi yang terbatas sehingga jenis kamera yang dapat digunakan terbatas dan waktu pemrosesan yang lama
2. *Frame per second* yang kecil, sehingga jika gerakan terlalu cepat, gerakan tidak akan tertangkap oleh kamera.
3. Sistem tidak dapat berjalan apabila Raspberry Pi terlalu panas, sehingga sistem harus diletakkan di tempat yang dingin.

1.5 Metodologi Penelitian

1. Studi Literatur

Pada tahap ini dilakukan pengumpulan dasar teori yang menunjang dalam penulisan Tugas Akhir. Dasar teori ini dapat diambil dari buku-buku, jurnal, dan artikel-artikel di internet dan forum-forum diskusi internet.

2. Perancangan Perangkat Keras

Perangkat keras kemudian dirancang terlebih dahulu. Perangkat keras yang akan digunakan adalah Raspberry Pi, Kamera, pengeras suara, perangkat untuk memasukkan data, dan *server*. Perangkat pelengkap lainnya untuk memrogram Raspberry Pi dapat bermacam-macam, contohnya digunakan *keyboard*, monitor, dan mouse.

3. Perancangan Perangkat Lunak

Perancangan perangkat lunak dilakukan. Perangkat lunak digunakan untuk mengintegrasikan kamera dan pengeras suara ke dalam Raspberry Pi, perancangan sistem pendeteksi gerakan, dan juga mengintegrasikan Raspberry Pi dan *server* yang menerima data.

4. Pengujian Perangkat Keras dan Perangkat Lunak

Pada tahap ini pengujian dilakukan untuk melihat karakteristik dan efisiensi dari sistem. Hasil pengujian ini kemudian akan dianalisa lebih lanjut.

5. Analisa

Analisa dilakukan terhadap hasil dari pengujian sehingga dapat ditentukan karakteristik dari sistem serta efisiensinya. Apabila karakteristik dan efisiensi belum sesuai dengan harapan maka pengujian kembali dilakukan.

6. Pembuatan Kemasan

Produk ini bertujuan agar dapat dipasarkan, maka dari itu setelah karakteristik dan efisiensi alat telah sesuai dengan harapan maka perlu dikemas semenarik mungkin. Hal ini bertujuan untuk menarik minat pasar.

7. Penyusunan Laporan

Proses terakhir adalah membuat dokumentasi pelaksanaan tugas akhir yang meliputi dasar teori, proses perancangan, pembuatan, dan pengujian aplikasi.

1.6 Sistematika Penulisan

Laporan tugas akhir ini terdiri dari lima bab dengan sistematika penulisan sebagai berikut:

- Bab 1 : Pendahuluan
Bab ini meliputi latar belakang, perumusan masalah, tujuan, sistematika penulisan, metodologi, dan relevansi.
- Bab 2 : Dasar Teori
Bab ini menjelaskan tentang berbagai macam teori-teori penunjang dalam pengerjaan tugas akhir ini, diantaranya : Raspberry Pi, *webcam*, *speaker*, jaringan WLAN, dan *database*.
- Bab 3: Perancangan Sistem
Pada bab ini menjelaskan tentang langkah-langkah dalam merencanakan dan membangun sistem secara keseluruhan baik *software* maupun *hardware* agar didapatkan sistem ini.
- Bab 4 : Pengujian dan Analisis
Bab ini menjelaskan data yang didapat dari pengujian keseluruhan sistem beserta analisisnya.
- Bab 5 : Penutup
Bagian ini merupakan bagian akhir yang berisikan kesimpulan yang diperoleh dari pembuatan Tugas Akhir ini, serta saran-saran untuk pengembangannya.

1.7 Relevansi

Hasil dari tugas akhir ini diharapkan dapat membantu pengembangan sistem monitorin yang terjangkau serta dapat membantu pengawasan ujian yang dilakukan di sekolah maupun universitas.

BAB 2

DASAR TEORI

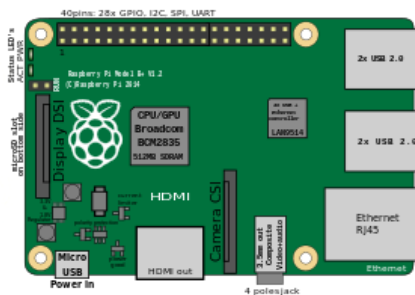
Pada bab ini dijelaskan teori-teori penunjang yang berhubungan dan membantu keseluruhan sistem pada Tugas Akhir ini.

2.1 Raspberry Pi

Raspberry Pi adalah komputer papan tunggal yang dibuat oleh sebuah yayasan bernama *Raspberry Pi Foundation* yang terletak di Inggris. Tujuannya ialah untuk mengajarkan dasar-dasar ilmu komputer di sekolah.

Raspberry Pi yang pertama berbasiskan *system on chip* (SoC) *Broadcom BCM2835*, yang disertai prosesor ARM1176JZF-S 700 MHz, GPU *VideoCore IV*, dan memiliki RAM 256 MB, yang kemudian dikembangkan menjadi model B dan B+ yang memiliki RAM 512 MB. Pada tahun 2015, *Raspberry Pi Foundation* mengeluarkan Raspberry Pi 2 yang memiliki prosesor ARM Cortex-A7 900MHz quad-core dan RAM 1 GB. Sebagai media penyimpanan, Raspberry Pi model A dan B memiliki soket SD, sedangkan untuk model A+, B+, dan 2 memiliki soket MicroSD.[1]

Raspberry Pi bersifat *open source* dengan basis Linux. Hal ini membuat Raspberry Pi dapat dimodifikasi sesuai kebutuhan penggunaannya. Sistem operasi utamanya menggunakan Linux Debian dengan bahasa pemrograman utama Python. Raspberry Pi juga mendukung bahasa C, C++, Java, Perl, dan Ruby.[1]



Gambar 2.1 Raspberry Pi B+ dan 2[2]

Berikut ini merupakan spesifikasi dari Raspberry Pi B+:

- Chip : Broadcom BCM2835 (CPU, GPU, DSP, SDRAM, dan satu port USB)
- CPU : 700 MHz ARM1176JZF-S core
- GPU : Broadcom VideoCore IV, OpenGL ES 2.0, 1080p30 h.264/MPEG-4 AVC *high-profile decoder*
- SDRAM : 512 MB
- Port USB 2.0 : 4 port
- Keluaran video : HDMI (rev 1.3 & 1.4), raw LCD Panels via DSI 14 HDMI resolusi dari 640×350 hingga 1920×1200.
- Keluaran audio : 3.5 mm jack, HDMI
- Penyimpanan : slot MicroSD
- Penghubung jaringan : 10/100 Ethernet (RJ45)
- *Low-level peripherals* : 8 × GPIO, UART, I²C bus, SPI bus, +3.3 V, +5 V, ground
- *Power ratings* : 600 mA (3.0 W)
- *Power source* : 5 volt menggunakan MicroUSB atau GPIO header
- Ukuran : 85.60 × 53.98 mm
- Berat : 45 gram

Berikut ini merupakan spesifikasi dari Raspberry Pi 2:

- Chip : Broadcom BCM2836 (CPU, GPU, DSP, SDRAM, dan satu port USB)
- CPU : 900 MHz quad-core ARM Cortex-A7
- GPU : Broadcom VideoCore IV, OpenGL ES 2.0, 1080p30 h.264/MPEG-4 AVC *high-profile decoder*
- SDRAM : 1 GB
- Port USB 2.0 : 4 port
- Keluaran video : HDMI (rev 1.3 & 1.4), raw LCD Panels via DSI 14 HDMI resolusi dari 640×350 hingga 1920×1200.
- Keluaran audio : 3.5 mm jack, HDMI
- Penyimpanan : slot MicroSD
- Penghubung jaringan : 10/100 Ethernet (RJ45)
- *Low-level peripherals* : 8 × GPIO, UART, I²C bus, SPI bus, +3.3 V, +5 V, ground
- *Power ratings* : 800 mA (4.0 W)

- *Power source* : 5 volt menggunakan MicroUSB atau GPIO header
- Ukuran : 85.60×53.98 mm
- Berat : 45 gram

Sistem operasi Raspberry Pi biasanya menggunakan sistem operasi berbasis Linux. Akan tetapi, Raspberry Pi 2 dapat menggunakan Windows 10. Salah satu sistem operasi yang banyak digunakan untuk Raspberry Pi ialah Raspbian. Sistem operasi ini dibuat berdasarkan arsitektur Debian 7 Wheezy. Sistem operasi lain yang dapat banyak digunakan ialah OpenELEC, Pidora, dan RISC OS, namun masih banyak lagi sistem operasi yang dapat digunakan di Raspberry Pi.[1]

2.2 Webcam

Kamera web ialah sejenis kamera yang dapat memberikan gambar secara *real time* (waktu nyata) menggunakan jaringan komputer menuju komputer, yang nantinya dapat disimpan, diperlihatkan, ataupun dikirimkan ke sistem atau perangkat lain menggunakan sistem jaringan seperti internet.[5] Kamera web biasanya dikoneksikan menggunakan kabel USB atau sejenisnya.



Gambar 2.2 Kamera Web[5]

Kamera web memiliki banyak kegunaan, diantaranya *videocalling* dan *videoconference* yang memungkinkan dua atau lebih orang saling berhubungan tatap muka walaupun tidak bertemu secara langsung. Fungsi ini sudah sangat marak digunakan karena sangat efisien dan efektif. Fungsi lainnya ialah dalam bidang kesehatan, karena telah ada

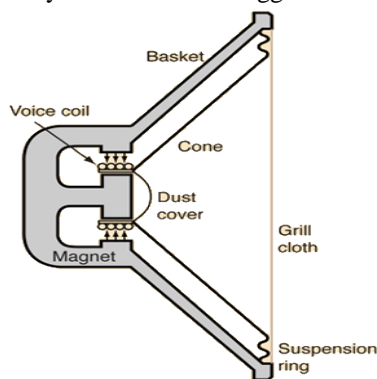
kamera web yang dapat menangkap detak arteri, yang diklaim memiliki akurasi plus minus 5 detak per menit.

Pengawasan dengan menggunakan video juga salah satu kegunaan kamera web yang penting. Pengawasan dapat dilakukan dengan berbagai alasan, diantaranya alasan keamanan, pengawasan anak-anak, pengawasan kantor yang dilakukan oleh atasan, dan lain-lain. Salah satu kegunaan kamera web yang mulai marak ialah sebagai pengawasan di dalam kelas. Selain dapat menambah keamanan kelas, juga dapat digunakan staf sekolah untuk mengawasi siswa-siswinya.

2.3 Speaker

Pengeras suara adalah sejenis transduser yang mengubah sinyal elektrik ke frekuensi audio dengan cara menggetarkan komponennya yang berbentuk membran untuk menggetarkan udara sehingga dapat menghasilkan suara yang dapat didengar. Pengeras suara dapat terbuat dari logam dan memiliki membran, kumparan, serta magnet.[4] Membran berfungsi untuk mengeluarkan suara. Fungsi pengeras suara secara keseluruhan adalah mengubah gelombang listrik dari perangkat penguat audio menjadi gelombang suara atau getaran.

Pada umumnya, pengeras suara terdiri dari tiga macam berdasarkan tinggi rendah bunyi yang dihasilkan. Pengeras suara dengan keluaran nada rendah biasa disebut *woofer*, pengeras suara yang menghasilkan bunyi vokal atau nada menengah disebut *midrange*, sedangkan dengan bunyi keluaran nada tinggi disebut *twitter*.[4]



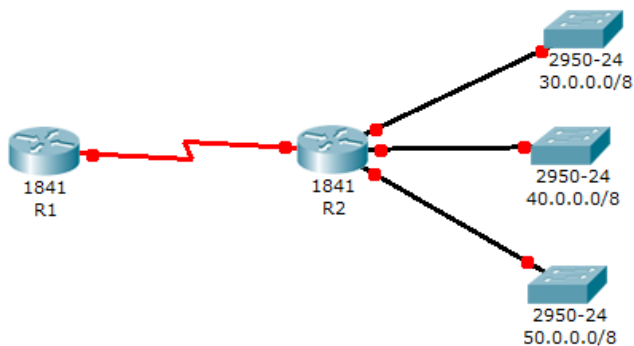
Gambar 2.3 Pengeras Suara Dinamis[4]

Tipe pengeras suara yang banyak digunakan ialah pengeras suara dinamis. Ketika sinyal audio yang berbentuk arus bolak-balik dimasukkan ke kumparan suara, kumparan kawat yang tergantung di celah melingkar diantara kutub magnet permanen, kumparan tersebut akan dipaksa bergerak maju mundur dikarenakan hukum induksi Faraday, yang kemudian menyebabkan diafragma yang terletak di kumparan bergerak maju mundur, mendorong udara untuk menghasilkan gelombang suara. Selain metode ini, terdapat beberapa teknologi alternatif yang dapat mengubah sinyal listrik menjadi suara.[4]

2.4 Router

Router adalah perangkat yang dapat meneruskan paket data antar jaringan komputer. *Router* terhubung dengan dua atau lebih barisan data dari jaringan yang berbeda. Ketika paket data dikirimkan menuju jaringan tertentu, router akan membaca alamat paket data tersebut yang kemudian akan diteruskan ke alamat yang dituju. *Router* banyak digunakan dalam jaringan berbasis teknologi protokol TCP/IP.[13]

Secara umum, *router* dibagi menjadi dua, yaitu *router* statis dan *router* dinamis. *Router* statis memiliki tabel *routing* statis yang diatur secara manual sehingga *router* ini mengenal semua alamat yang ada di kaki-kakinya. *Router* jenis ini cocok digunakan untuk jaringan kecil. Pengiriman data di *router* ini pun lebih cepat karena telah mengetahui alamat yang ada sehingga jalur dapat dengan mudah ditentukan.[13]



Gambar 2.4 Sistem Router Statis[13]

Router dinamis memiliki dan membuat tabel *routing* dinamis, dengan mendengarkan lalu lintas jaringan dan juga dengan saling berhubungan dengan *router* lainnya. *Router* ini cocok digunakan pada jaringan yang besar. *Router* ini tidak perlu mengetahui semua alamat, melainkan hanya alamat *router* kakinya.[13]

2.5 WLAN

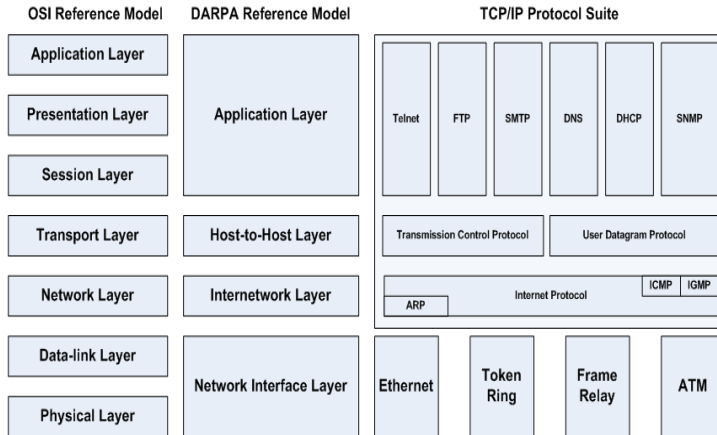
Wireless local area network adalah jaringan komputer nirkabel yang menghubungkan dua atau lebih perangkat menggunakan metode distribusi nirkabel di dalam area terbatas.[14] Hal ini memungkinkan pengguna berpindah tempat dan tetap terhubung ke jaringan, selama masih di dalam cakupan wilayah. Media yang dapat digunakan sebagai transmisi adalah gelombang radio, gelombang mikro, dan cahaya inframerah. WLAN yang paling sering digunakan didasarkan oleh standar IEEE 802.11 yang dipasarkan dengan merek dagang Wi-Fi.[14]

2.6 TCP/IP

TCP/IP (*Transmission Control Protocol/Internet Protocol*) adalah gabungan dari protokol TCP (*Transmission Control Protocol*) dan IP (*Internet Protocol*) sebagai sekelompok protokol yang mengatur komunikasi data dalam proses tukar menukar data dari satu komputer ke komputer lain di dalam jaringan internet yang akan memastikan pengiriman data sampai ke alamat yang dituju.[15]

Protokol ini menggunakan skema pengalamatan yang sederhana, yaitu *IP address* yang memungkinkan berjuta komputer dapat berhubungan satu sama lain di internet. Protokol ini juga cocok untuk menghubungkan sistem-sistem berbeda untuk membentuk jaringan heterogen.[15]

Arsitektur TCP/IP tidak berbasis tujuh lapisan OSI, melainkan menggunakan model referensi DARPA. TCP/IP mengimplementasikan arsitektur berlapis terdiri atas 4 lapis. Empat lapis ini dapat dipetakan terhadap model referensi OSI.[15]



Gambar 2.5 Perbandingan Model Referensi DARPA dan OSI[15]

Protokol utama dalam protokol TCP/IP adalah sebagai berikut:

- Protokol lapisan aplikasi yang bertanggung jawab untuk menyediakan akses kepada aplikasi terhadap layanan jaringan TCP/IP.
- Protokol lapisan antar-host yang berguna untuk membuat komunikasi.
- Protokol lapisan *internetwork* yang bertanggung jawab melakukan pemetaan dan enkapsulasi paket-paket data jaringan menjadi paket-paket IP.
- Protokol lapisan antarmuka jaringan yang bertanggung jawab untuk meletakkan frame-frame jaringan di atas media jaringan yang digunakan.

2.7 OpenCV

OpenCV adalah *open source library* yang mencakup beberapa ratus algoritma pengolahan citra secara *real time*. OpenCV menggunakan lisensi BSD, sehingga OpenCV bebas digunakan baik untuk kepentingan edukasi, maupun kepentingan komersial. OpenCV dikembangkan oleh pusat riset Intel di Russia.[10]



Gambar 2.6 Logo OpenCV[9]

OpenCV diluncurkan pada tahun 1999 oleh pusat riset Intel Russia. Pada awalnya tujuan dari proyek OpenCV ini adalah riset pengolahan citra lebih lanjut dan menyajikan infrastruktur dasarnya sehingga pengembangan akan menjadi lebih cepat.

OpenCV memiliki tatap muka untuk bahasa C, C++, Python, dan Java. OpenCV juga mendukung perangkat Windows, Linux, Android, iOS, dan Mac OS. OpenCV memiliki banyak area aplikasi, seperti pengenalan wajah, pengenalan gerakan, *motion tracking*, interaksi manusia komputer, *stereo vision*, dan sebagainya.[9]

2.8 Python

Python adalah bahasa pemrograman tingkat tinggi dan berorientasi objek dengan semantik dinamis. Python sangat berguna untuk pengembangan aplikasi karena struktur datanya yang dibangun dengan level yang tinggi digabung dengan pengetikan dan penggabungan yang dinamis.[16]



Gambar 2.7 Logo Python[16]

Python dapat dijalankan di berbagai platform sistem operasi, contohnya Linux, Windows, Mac OS X, Amiga, Palm, dan Symbian. Python didistribusikan dengan beberapa lisensi yang berbeda dari beberapa versi, namun pada prinsipnya Python dapat dipergunakan secara bebas, bahkan dikomersialkan.[16]

Fitur-fitur yang dimiliki Python antara lain:

- memiliki kepustakaan yang luas; dalam distribusi Python telah disediakan modul-modul 'siap pakai' untuk berbagai keperluan.
- berorientasi obyek.
- memiliki sistem pengelolaan memori otomatis (garbage collection, seperti java)
- modular, mudah dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa Python maupun C/C++.
- memiliki banyak fasilitas pendukung sehingga mudah dalam pengoprasiaannya.

2.9 Web Server Apache

Web server merupakan software yang memberikan layanan data yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan browser web dan mengirimkan kembali hasilnya dalam bentuk halaman - halaman web yang umumnya berbentuk dokumen HTML.[6] Beberapa macam Web Server diantaranya:

1. Apache
2. Microsoft windows Server 2003 Internet Information Services (IIS)
3. Lighttpd
4. Sun Java System Web Server
5. Xitami Web Server
6. Zeus Web Server
7. Nginx



Gambar 2.8 Logo Web Server Apache[3]

Apache merupakan web server yang paling banyak dipergunakan di Internet.[6] Program ini pertama kali didesain untuk sistem operasi lingkungan UNIX. Namun demikian, pada beberapa versi berikutnya Apache mengeluarkan programnya yang dapat dijalankan di Windows NT. Apache mempunyai program pendukung yang cukup banyak. Hal ini memberikan layanan yang cukup lengkap bagi penggunaanya.

2.10 Database MySql

Database adalah sebuah system yang di buat untuk mengorganisasi, menyimpan dan menarik data dengan mudah.[12] Database terdiri dari kumplan data yang terorganisir untuk 1 atau lebih penggunaan, dalam bentuk digital. Database digital di manage menggunakan Database Management System (DBMS), yang menyimpan isi database, mengizinkan pembuatan dan maintenance data dan pencarian dan akses yang lain.[12] Beberapa Database yang ada saat ini adalah : Mysql, Sql Server, Ms.Access, Oracle, dan PostgreSql.



Gambar 2.9 Logo MySQL[3]

MySQL adalah salah satu jenis database server yang paling umum digunakan dan dikembangkan oleh *Oracle Corporation*. MySQL menggunakan bahasa SQL untuk mengakses database nya. Lisensi Mysql adalah FOSS License Exception dan ada juga yang versi komersial nya. MySQL tersedia untuk beberapa platform, di antara nya adalah untuk versi windows dan versi linux. MySQL memiliki beberapa kelebihan yaitu, bersifat *open source*, dapat dijalankan di banyak platform, dan dapat diintegrasikan dengan beberapa bahasa lain seperti Net, Java Python, dan Perl.[3]

2.11 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman web yang dapat disisipkan ke dalam bahas HTML dan banyak digunakan untuk pengembangan situs web.[3] PHP dikembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh *The PHP Group*. PHP disebut bahasa pemrograman *server side* karena PHP diproses pada komputer server. PHP memiliki 8 tipe data, yaitu:

1. *Boolean*
2. *Integer*
3. *Float/Double*
4. *String*
5. *Array*
6. *Object*
7. *Resource*
8. *NULL*

PHP memiliki beberapa kelebihan dibandingkan bahasa pemrograman web lainnya, antara lain:

1. PHP adalah bahasa *script* yang tidak melakukan kompilasi dalam penggunaannya.
2. *Web Server* yang mendukung PHP banyak, contohnya apache, IIS, Lighttpd, dan Xitami, dengan konfigurasi yang relatif mudah.
3. PHP memiliki banyak pengembang, sehingga pengembangannya relatif mudah.
4. PHP adalah bahasa *open source* yang dapat dijalankan di berbagai mesin, seperti Linux, Unix, Macintosh, dan Windows.

Halaman ini sengaja dikosongkan

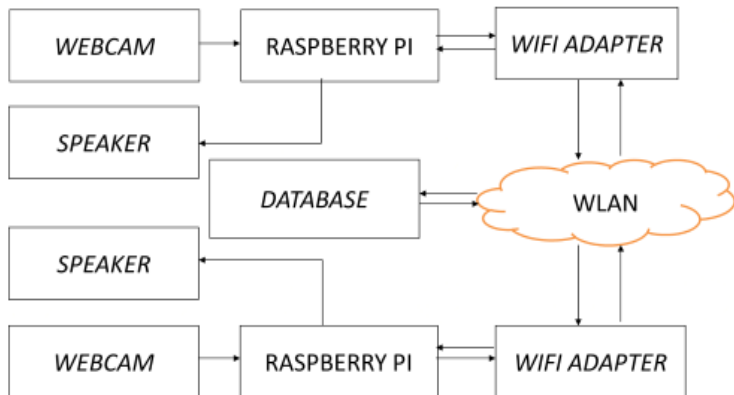
BAB 3

PERANCANGAN SISTEM

Pada bab ini akan dijelaskan perancangan sistem secara keseluruhan. Hal ini termasuk perancangan perangkat lunak dan perancangan perangkat keras. Kamera web dan *speaker* diletakkan di dalam kelas. Kamera digunakan untuk menangkap suasana kelas, sedangkan *speaker* digunakan untuk mengirimkan pengumuman yang dapat dikirim menggunakan komputer *server*. Pengiriman gambar dan penerimaan suara dilakukan oleh Raspberry Pi. Raspberry Pi berkomunikasi dengan *server* melalui WLAN.

3.1 Diagram Blok

Cara kerja sistem ini secara umum dapat digambarkan dalam blok diagram ini.



Gambar 3.1 Diagram Blok

Webcam akan menangkap gambar, kemudian akan mengirimkannya secara *real time* menuju Raspberry Pi. Raspberry Pi kemudian mengirimkan gambar-gambar tersebut menuju *server*.

Pengiriman gambar menuju server menggunakan jaringan WLAN (*Wireless Local Area Network*). Gambar tersebut kemudian ditampilkan di *server*.

Pengiriman suara dilakukan oleh *server*. Pertama, suara akan ditangkap menggunakan *microphone* yang ada pada *server*. Data kemudian dikirimkan menuju Raspberry dengan menggunakan WLAN. Setelah data diterima di Raspberry Pi, data kemudian akan diteruskan menuju *speaker* yang kemudian akan diubah menjadi suara.

3.2 Router dan Wifi Adapter

3.2.1 Router TP-Link TL-MR3220

Router yang digunakan adalah TP-Link MR3220. *Router* ini dipilih atas pertimbangan durabilitas dan harga yang murah jika dibandingkan dengan fitur yang didapat. *Router* ini menggunakan sumber tegangan 9 VDC yang berasal dari adapter yang mengambil sumber dari tegangan 220 VAC. *Router* ini nantinya akan menjadi penghubung antara *server* dan Raspberry Pi.



Gambar 3.2 TP-Link TL-MR3220[18]

Berikut ini adalah spesifikasinya :

Hardware Features Interfaces

USB 2.0 Port for LTE/HSPA +/HSUPA
/HSDPA/UMTS/EVDO USB Modem

	1 10/100Mbps WAN Port, 4 10/100Mbps LAN Ports, support the auto-Negotiation and auto-MDI/MDIX
<i>Button</i>	WPS/Reset Button Wireless On/Off Switch Power On/Off Button
<i>Power Supply</i>	9VDC/0.85A
<i>Dimensions</i>	204x138x44mm
<i>Antenna Type</i>	Omni directional, Detachable Reverse SMA
<i>Antenna Gain</i>	5dBi

Wireless Features

<i>Wireless Standards</i>	IEEE 802.11n*, IEEE 802.11g, IEEE 802.11b
<i>Frequency</i>	2.4-2.4835GHz
<i>Transmit Power</i>	<20dBm
<i>Wireless Security</i>	Support 64/128 bit WEP, WPA-PSK/WPA2-PSK, Wireless MAC Filtering
<i>Modulation</i>	DBPSK, DQPSK, CCK, OFDM, 16-QAM, 64-QAM

Software Features

<i>Security</i>	NAT Firewall, SPI Firewall, MAC / IP / Packet / Application / URL Filtering, Denial of Service(DoS), SYN Flooding, Ping of Death
<i>Management</i>	Web Based Configuration(HTTP), Web Based Firmware Upgrade

Others

<i>Certification</i>	CE, FCC, RoHS
<i>Package Content</i>	3G/4G Wireless N Router TL-MR3220 1 Antenna Power supply unit Ethernet Cable Resource CD Quick Installation Guide
<i>System Requirments</i>	Microsoft® Windows® 98SE, NT, 2000, XP, Vista™ or Windows 7, MAC® OS, NetWare®, UNIX® or Linux.

<i>Environment</i>	<i>Operating Temperature:</i>	0°C~40°C
	<i>Storage Temperature:</i>	-40°C~70°C
	<i>Operating Humidity:</i>	10%~90% non-condensing
	<i>Storage Humidity:</i>	5%~90% non-condensing

3.2.2 Wifi Adapter TP-Link WN725N

Wifi adapter yang digunakan sebagai penghubung Raspberry Pi dan jaringan *wifi* adalah TP-Link WN725N. *Wifi Adapter* ini digunakan karena kompatibilitasnya dengan Raspberry Pi. Hanya sedikit *Wifi adapter* yang kompatibel dengan Raspberry Pi, dan TP-Link WN725N adalah salah satunya. *Driver wifi adapter* ini dapat langsung diunduh dari internet, dan masing-masing sistem operasi memiliki *driver* yang berbeda.



Gambar 3.3 TP-Link WN725N[17]

Berikut ini adalah spesifikasinya:

Hardware Features

<i>Interface</i>	USB 2.0
<i>LED</i>	Status
<i>Dimensions</i>	18.6x15x7.1mm
<i>Weight</i>	2.1 grams
<i>Antenna</i>	Internal antenna

Wireless Features

<i>Frequency</i>	2.400-2.435GHz
<i>Wireless Standards</i>	IEEE 802.11b, IEEE 802.11g, IEEE 802.11n

<i>Signal Rate</i>	11b: Up to 11Mbps (dynamic) 11g: Up to 54Mbps (dynamic) 11n: Up to 150Mbps (dynamic)
<i>Reception Sensitivity</i>	130M: -68dBm@10% PER 108M: -68dBm@10% PER 54M: -68dBm@10% PER 11M: -85dBm@8% PER 6M: -88dBm@10% PER 1M: -90dBm@8% PER
<i>Transmit Power</i>	<20dBm
<i>Wireless Modes</i>	Ad-Hoc / Infrastructure mode
<i>Wireless Security</i>	Supports 64/128 WEP, WPA/WPA2, WPA-PSK/WPA2-PSK (TKIP/AES), supports IEEE 802.1X
<i>Modulation</i>	DBPSK, DQPSK, CCK, OFDM, 16-QAM, 64-QAM
<i>Others</i>	
<i>Certification</i>	CE, FCC, IC, RoHS
<i>Package Content</i>	150Mbps wireless N Nano USB adapter TL-WN725N Resource CD Quick Installation Guide
<i>System Requirements</i>	Windows 8(32/64bits), Windows 7(32/64bits), Windows Vista(32/64bits), Windows XP(32/64bits)
<i>Environment</i>	Operating Temperature: 0°C~40°C Storage Temperature: -40°C~70°C Operating Humidity: 10%~90% non-condensing Storage Humidity: 5%~90% non-condensing

3.3 Perancangan Database

MySQL digunakan sebagai database dengan Apache sebagai web server. Database digunakan untuk penyimpanan gambar yang

nantinya akan digunakan untuk ditampilkan dari *server* sebagai *live steaming* dari kamera.

Perancangan dimulai dari pembuatan *database* di *localhost*. *Database* yang telah dibuat dinamakan “raspberypi” yang terdapat pada *server* 127.0.0.1.

Database	Collation	
<input type="checkbox"/> cdcol	latin1_general_ci	Check Privileges
<input type="checkbox"/> information_schema	utf8_general_ci	Check Privileges
<input type="checkbox"/> mysql	latin1_swedish_ci	Check Privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	Check Privileges
<input type="checkbox"/> phpmyadmin	utf8_bin	Check Privileges
<input type="checkbox"/> raspberypi	latin1_swedish_ci	Check Privileges
<input type="checkbox"/> webauth	latin1_general_ci	Check Privileges
Total: 7	latin1_swedish_ci	

Gambar 3.4 *Database* “raspberypi”

Setelah pembuatan *database*, maka selanjutnya dibuat tabel yang akan diisi data yang disimpan. Tabel berada didalam *database*. Tabel yang dibuat dinamakan “upload”.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> demo	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
<input type="checkbox"/> upload	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 K1B	-
2 tables	Sum	9	InnoDB	latin1_swedish_ci	32 K1B	0 B

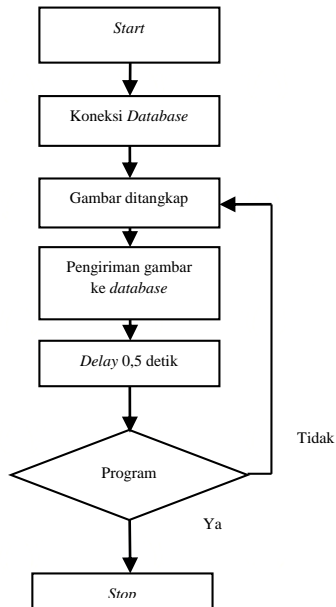
Gambar 3.5 Tabel “upload”

3.4 Perancangan Pengiriman Gambar

Pengiriman gambar dilakukan dengan menggunakan OpenCV. Pengiriman gambar menuju *server* dilakukan satu per satu. *Frame per second* yang digunakan ialah 2, artinya dalam 1 detik terdapat 2 gambar yang dikirim menuju *server*.

Program dirancang dengan menggunakan bahasa Python. Program akan menampilkan video, menangkap gambar, kemudian menyimpannya ke dalam *server*. Proses ini berlangsung selama kurang

lebih 0,5 detik dan akan diulang terus menerus. Hasilnya, akan terdapat banyak gambar yang tersimpan di dalam *server*. Gambar ini kemudian akan ditampilkan secara bergantian sehingga menjadi video.

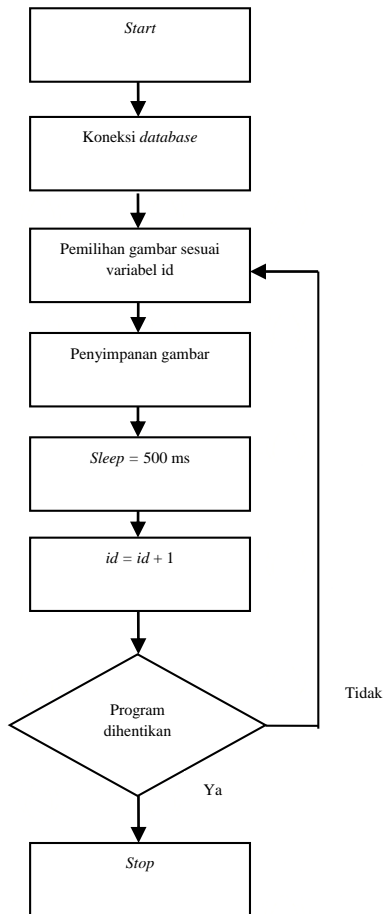


Gambar 3.6 *Flowchart* Pengiriman Gambar

Program diawali dengan pemunculan video. Kemudian, setiap *frame* akan ditangkap sehingga menjadi gambar. Gambar akan langsung dikirim menuju *database* yang telah disiapkan. Delay dihitung berdasarkan kemampuan Raspberry Pi memproses gambar. apabila program diteruskan, maka program akan kembali mengambil *frame*.

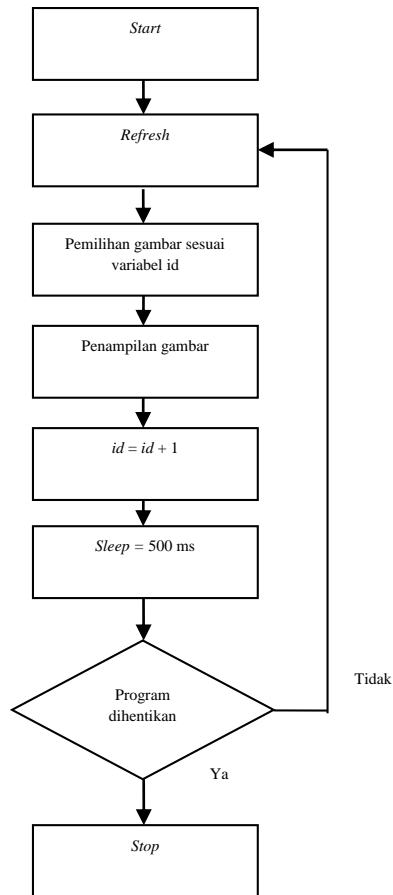
3.5 Perancangan Penampilan Gambar

Gambar-gambar yang telah tersimpan di *database* harus dapat ditampilkan di *server*. Perancangan penampilan gambar terdiri dari dua, yaitu perancangan *download* data gambar dan perancangan penampilan gambar.



Gambar 3.7 Flowchart Download Gambar

Proses *download* data gambar dimulai dari pemeriksaan dan penyesuaian nomor *id* dengan variabel yang telah disiapkan. Kemudian gambar dipilih sesuai dengan nomor *id* yang ada. Gambar yang telah dipilih kemudian disimpan di *folder* yang telah disiapkan. Program kemudian tidur selama 500 ms. Variabel kemudian ditambah 1. Apabila program diteruskan, maka variabel yang telah ditambah 1 disesuaikan lagi dengan *id* gambar.



Gambar 3.8 Flowchart Penampilan Gambar

Proses penampilan gambar dimulai dari pemilihan variabel dan penyesuaian gambar. Setelah disesuaikan, gambar yang memiliki nama variabel yang sama akan ditampilkan. Setelah ditampilkan, variabel akan ditambah satu. Program kemudian akan tidur selama 500 ms. Apabila program diteruskan, maka program akan melakukan *refresh* dan variabel baru akan disesuaikan lagi dengan gambar.

```

ini_set('max_execution_time', 200);
$con=mysql_connect("127.0.0.1","root","");
$db=mysql_select_db("raspberrypi",$con);
$id = 7563;
while (true == true){
    $qry="SELECT image FROM image WHERE id=$id";

    $result=mysql_query($qry);
    $fp=fopen(($id).".png","w") or die("unable to open file");
    $row = mysql_fetch_row($result);
    fwrite($fp,$row[0]);
    $id = $id+1;

    usleep(500000);
}

```

Gambar 3.9 Script Download Gambar

Script yang ditunjukkan pada gambar 3.9 adalah script yang digunakan untuk melakukan *download* gambar dari *database*. *Mysql_connect* digunakan untuk mengakses *localhost*, sedangkan *mysql_select_db* digunakan untuk memilih database yang ingin diakses.

Variabel *\$qry* adalah variabel yang digunakan untuk menentukan kondisi data yang diambil dari *query*. Untuk mengambil datanya, menggunakan perintah *mysql_query*. Perintah *fopen* digunakan untuk membuka data dan memberi nama. *Mysql_fetch_row* digunakan untuk menentukan tipe data yang diambil, sedangkan perintah *fwrite* digunakan untuk menulisnya ke dalam *folder*.

3.6 Perancangan Pengiriman Suara

Pengiriman suara dilakukan oleh *Pulseaudio*. *Pulseaudio* pertama diinstall terlebih dahulu. Kemudian *pulseaudio* diatur agar dapat tersambung dengan komputer *server*.


```

### Automatically move streams to the default sink if the sink they are
### connected to dies, similar for sources
load-module module-rescue-streams

### Make sure we always have a sink around, even if it is a null sink.
load-module module-always-sink

### Automatically suspend sinks/sources that become idle for too long
load-module module-suspend-on-idle

### Enable positioned event sounds
load-module module-position-event-sounds

load-module module-native-protocol-tcp auth-ip-acl=127.0.0.1;192.168.0.101/15

load-module module-zeroconf-publish

```

Gambar 3.10 Pengaturan *Pulseaudio*

Pengaturan dilakukan dengan penambahan dua baris dibawah. Penambahan dua baris tersebut berfungsi untuk meregistrasi Raspberry Pi pada *server*.

Pada komputer *server*, Raspberry Pi disambungkan dengan menggunakan Plink. Plink adalah program tatap muka (*interface*) yang menggunakan *command line* melalui Putty. *Command line* yang dimasukkan adalah :

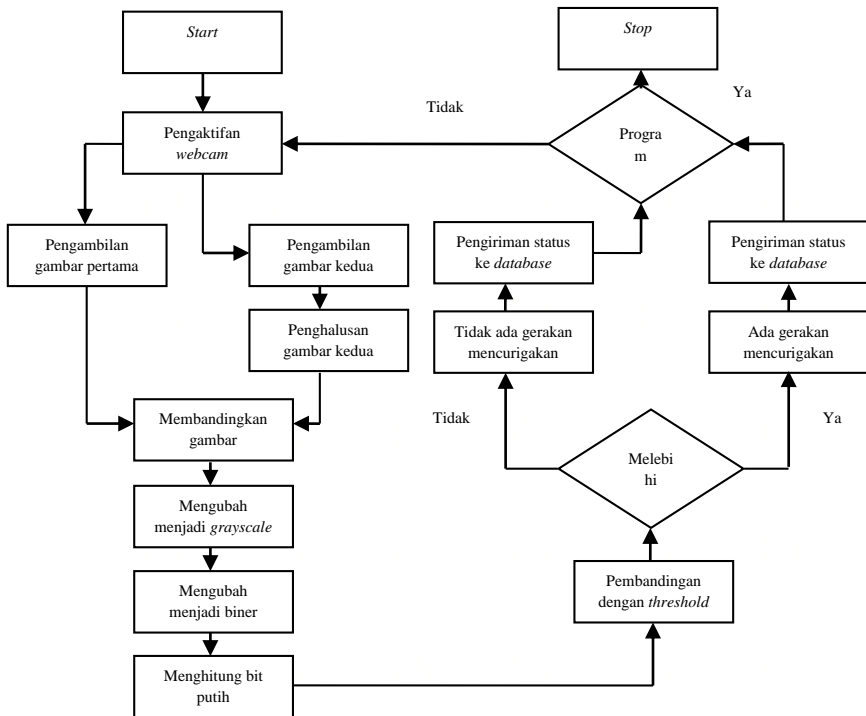
linco.exe -B 16 -C 2 -R 44100 | plink 192.168.0.103 -l pi -pw raspberry
"cat - | pacat --server 127.0.0.1 --playback"

yang artinya Raspberry Pi memiliki alamat 192.168.0.103, memiliki *username* pi dan *password* raspberry, juga menggunakan *server* 127.0.0.1.

3.7 Perancangan Pendeteksi Gerakan

Pendeteksi gerakan digunakan sebagai basis pendeteksi kecurangan dengan parameter gerakan ekstrim. Gerakan yang berlebihan seringkali dilakukan saat seseorang hendak melakukan kecurangan. Dengan demikian, sistem ini diharapkan dapat meningkatkan efektifitas dan efisiensi pengawasan.

Metode yang digunakan untuk mendeteksi gerakan adalah *background subtraction*. Metode ini bekerja dengan cara membandingkan antara dua gambar dan mencari perbedaannya. Perbedaan inilah yang nantinya akan dideteksi.



Gambar 3.11 Flowchart Pendeteksi Gerakan

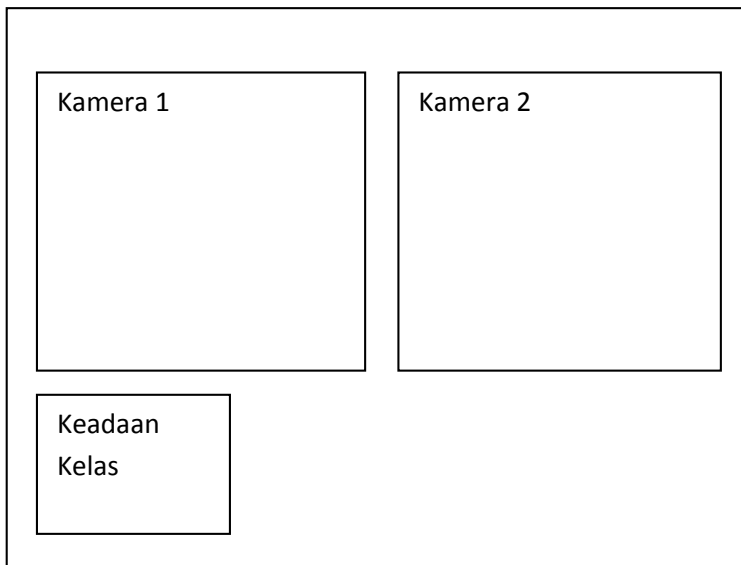
Pada saat program diaktifkan, *webcam* akan diaktifkan. Gambar kemudian diambil dua kali. Pengambilan gambar pertama akan langsung diteruskan menuju proses berikutnya, sedangkan pengambilan gambar kedua akan dihaluskan terlebih dahulu. Kemudian, kedua gambar dibandingkan dan dicari perbedaannya. Gambar yang telah dibandingkan kemudian diubah menjadi *greyscale*. Kemudian gambar *greyscale* diberi batasan sehingga menjadi gambar biner. Ketika ada gerakan maka bit bernilai 1, atau bit putih, akan menjadi semakin banyak. Bit putih inilah yang akan dijadikan parameter. Bit putih kemudian dihitung dan dibandingkan dengan *threshold* yang telah dipersiapkan. Apabila bit putih melebihi *threshold*, maka terjadi banyak gerakan. Apabila bit putih tidak

melebihi *threshold*, maka tidak terjadi banyak gerakan. Status ini kemudian dikirim menuju *database* dan diproses pada server.

3.8 Perancangan *Webpage*

Webpage dirancang sebagai tatap muka utama pada saat pengawasan. *Webpage* dirancang dengan menggunakan bahasa PHP. *Webpage* akan terdiri dari pengiriman gambar dari kedua kamera dan kondisi kelas yang dikirimkan oleh Raspberry Pi dan diambil dari *database*.

Webpage yang saya buat terdiri dari penampilan gambar dari kedua kamera, dan juga keadaan kelas saat ujian.



Gambar 3.12 Desain *Webpage*

Gambar 3.12 menunjukkan perancangan desain *webpage*. Kamera 1 akan menunjukkan keadaan kelas pertama, sedangkan Kamera 2 akan menunjukkan keadaan kelas kedua. Keadaan Kelas menunjukkan apakah didalam kelas tertentu terdapat gerakan yang mencurigakan atau tidak.

```

<table width="730" border="1">
  <tr>
    <th colspan="3" scope="col"><div align="center">AKTIVITAS KELAS</div></th>
  </tr>
  <tr>
    <td width="365"><div align="center">
      <?php
        include 'id.php';
        echo $id;
        $id=$id+1;
        $img= ($id).".png" ;
        $image = '';
        echo($image);
      ?>
    </div></td>
    <td height="200" colspan="2"><div align="center">
      <?php
        include 'id.php';
        echo $id;
        $id=$id+1;
        $img= ($id).".png" ;
        $image = '';
        echo($image);
      ?>
    </div></td>
  </tr>
</table>

```

Gambar 3.13 *Script* Penampilan Gambar

Script yang ditunjukkan pada gambar 3.13 digunakan untuk menampilkan gambar dari kedua kamera. Perintah yang digunakan untuk menampilkan gambar adalah *img src*, diikuti dengan sumber dan luas *window*. *\$id* adalah variabel yang digunakan untuk menandakan gambar yang ditampilkan.

```

<?php
include 'id.php';
$con=mysql_connect("127.0.0.1","root","");
$db=mysql_select_db("rasberryui",$con);
$qry="SELECT status FROM image WHERE id=$id";
$result=mysql_query($qry);
$row = mysql_fetch_array($result, MYSQL_ASSOC);
$fromid = $row['status'];
echo $fromid;
$id=$id+1;
$var_str = var_export($id, true);
$var = "<?php\n\n$id = $var_str;\n\n?>";
file_put_contents('id.php',$var);
?>

```

Gambar 3.14 *Script* Penampilan Status

Script yang ditampilkan pada gambar 3.14 digunakan untuk menampilkan status kelas tertentu. Untuk melakukan koneksi ke *database* menggunakan perintah *mysql_connect* dan *mysql_select_db*. Perintah *mysql_connect* digunakan untuk melakukan koneksi ke *localhost*, sedangkan *mysql_select_db* digunakan untuk memilih *database* yang digunakan, dalam hal ini adalah 'raspberrypi'. Variabel \$qry adalah variabel yang digunakan untuk memberikan kondisi untuk mengambil data dari *query* tertentu.

Perintah yang digunakan untuk mengambil data dari *query* adalah *mysql_query* dan *mysql_fetch_array*. Perintah *mysql_query* digunakan untuk mengambil data dari *query* tertentu, sedangkan *mysql_fetch_array* digunakan untuk menentukan tipe data yang saat pengambilan. Perintah yang digunakan untuk mengeluarkan status adalah *echo*.

Halaman ini sengaja dikosongkan

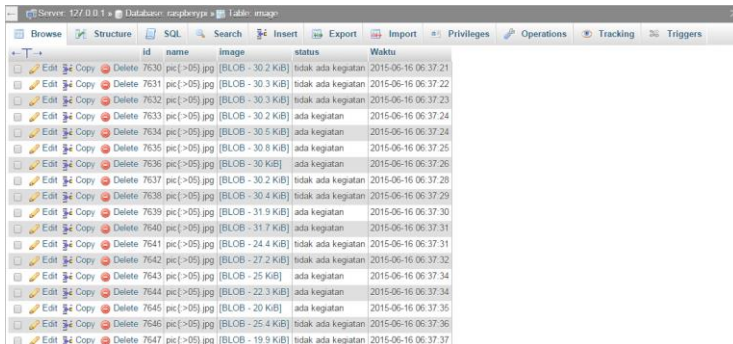
BAB 4

PENGUJIAN DAN ANALISIS

Pada bab ini akan dilakukan pengujian dan analisis sistem. Pengujian dan analisis dilakukan pada sistem dengan tujuan mengetahui efektifitas dan cara kerja sistem. Pengujian juga dilakukan untuk mengetahui batasan-batasan sistem.

4.1 Pengujian Pengiriman Gambar

Pengujian ini bertujuan untuk mengetahui apakah pengiriman gambar menuju *database* berhasil. Pengujian ini dilakukan dengan menjalankan program pengiriman gambar di Raspberry Pi dan menjalankan *database*. Pengujian berhasil apabila data gambar berhasil tersimpan di dalam *database*.



	id	name	image	status	Waktu
Edit Copy Delete	7630	pic(-05).jpg	(BLOB - 30.2 KiB)	tidak ada kegiatan	2015-06-16 06:37:21
Edit Copy Delete	7631	pic(-05).jpg	(BLOB - 30.3 KiB)	tidak ada kegiatan	2015-06-16 06:37:22
Edit Copy Delete	7632	pic(-05).jpg	(BLOB - 30.3 KiB)	tidak ada kegiatan	2015-06-16 06:37:23
Edit Copy Delete	7633	pic(-05).jpg	(BLOB - 30.2 KiB)	ada kegiatan	2015-06-16 06:37:24
Edit Copy Delete	7634	pic(-05).jpg	(BLOB - 30.5 KiB)	ada kegiatan	2015-06-16 06:37:24
Edit Copy Delete	7635	pic(-05).jpg	(BLOB - 30.8 KiB)	ada kegiatan	2015-06-16 06:37:25
Edit Copy Delete	7636	pic(-05).jpg	(BLOB - 30 KiB)	ada kegiatan	2015-06-16 06:37:26
Edit Copy Delete	7637	pic(-05).jpg	(BLOB - 30.2 KiB)	tidak ada kegiatan	2015-06-16 06:37:28
Edit Copy Delete	7638	pic(-05).jpg	(BLOB - 30.4 KiB)	tidak ada kegiatan	2015-06-16 06:37:29
Edit Copy Delete	7639	pic(-05).jpg	(BLOB - 31.9 KiB)	ada kegiatan	2015-06-16 06:37:30
Edit Copy Delete	7640	pic(-05).jpg	(BLOB - 31.7 KiB)	ada kegiatan	2015-06-16 06:37:31
Edit Copy Delete	7641	pic(-05).jpg	(BLOB - 24.4 KiB)	tidak ada kegiatan	2015-06-16 06:37:31
Edit Copy Delete	7642	pic(-05).jpg	(BLOB - 27.2 KiB)	tidak ada kegiatan	2015-06-16 06:37:32
Edit Copy Delete	7643	pic(-05).jpg	(BLOB - 25 KiB)	ada kegiatan	2015-06-16 06:37:34
Edit Copy Delete	7644	pic(-05).jpg	(BLOB - 22.3 KiB)	ada kegiatan	2015-06-16 06:37:34
Edit Copy Delete	7645	pic(-05).jpg	(BLOB - 20 KiB)	ada kegiatan	2015-06-16 06:37:35
Edit Copy Delete	7646	pic(-05).jpg	(BLOB - 25.4 KiB)	tidak ada kegiatan	2015-06-16 06:37:36
Edit Copy Delete	7647	pic(-05).jpg	(BLOB - 19.9 KiB)	tidak ada kegiatan	2015-06-16 06:37:37

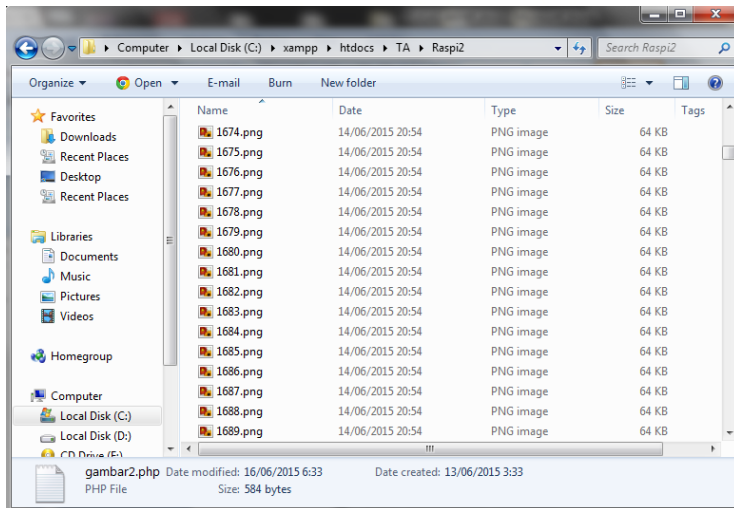
Gambar 4.1 *Database* Pengiriman Gambar

Ketika program dijalankan, gambar langsung tersimpan satu per satu. Kolom *id* menunjukkan nomor gambar yang tersimpan, sedangkan kolom *image* menunjukkan kolom dimana gambar tersimpan. Tipe data gambar saat tersimpan adalah biner (.bin).

Setelah pengujian dilakukan dapat disimpulkan bahwa pengiriman gambar berhasil dilakukan. *Delay* pengiriman gambar terjadi ketika koneksi terganggu, akan tetapi apabila koneksi terus terjaga kualitasnya, maka pengiriman gambar tidak akan terganggu.

4.2 Pengujian Penampilan Gambar

Pengujian ini bertujuan untuk memeriksa apakah gambar berhasil tersimpan di dalam *folder* dan gambar dapat diambil dari suatu *folder*. Pengujian dilakukan dengan cara menjalankan *script* PHP yang telah disediakan.



Gambar 4.2 *Folder* Penyimpanan Gambar

Setelah pengujian dilakukan, variabel *\$id* yang telah dipersiapkan pada *script* dicocokkan pada kolom *id* yang ada pada *database*. Setelah dicocokkan, maka *file* yang ada pada *database* akan langsung tersimpan di dalam *folder* dimana *script* tersimpan.

Karena penyimpanan gambar dilakukan oleh *server*, maka performa Raspberry Pi tidak berpengaruh terhadap penyimpanan gambar. Ketika *script* dinyalakan, maka data langsung tersimpan secara otomatis dengan *delay* yang telah ditentukan.

Pengujian penampilan gambar dilakukan dengan cara menjalankan *script* yang telah disediakan dan melihat apakah kondisi ruangan tertampilkkan.



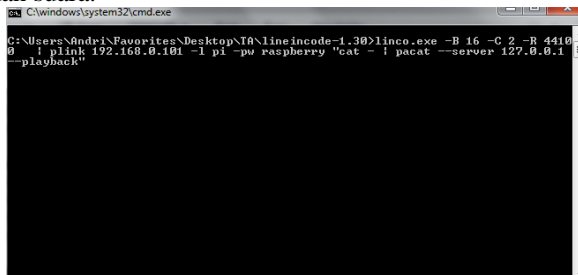
Gambar 4.3 Hasil Penampilan Gambar

Setelah *script* dijalankan, gambar yang tersimpan di dalam *folder* akan langsung ditampilkan yang disesuaikan dengan variabel *\$id* yang tersedia. Gambar akan muncul secara bergantian dengan *delay* yang telah ditentukan sebelumnya, yaitu 500 ms. Akan tetapi, setiap perpindahan gambar terjadi kedipan.

Pergantian gambar dilakukan dengan cara memutar ulang (*refresh*) *webpage* yang ada. Setiap *refresh*, gambar akan berganti menjadi gambar selanjutnya. Hal ini menyebabkan terjadinya kedipan setiap pergantian gambar.

4.2 Pengujian Pengiriman Suara

Pengujian ini bertujuan untuk mengetahui apakah pengiriman suara berhasil dan memperhatikan *delay* yang ada setiap pengiriman suara. *Delay* akan ditentukan dari lamanya data suara sampai di tujuan. Pengujian juga dilakukan untuk mengetahui hal-hal yang mempengaruhi pengiriman suara.



Gambar 4.4 Tampilan Pengiriman Suara

Delay akan didapat dari rata-rata yang diambil dari lima sampel dari masing-masing Raspberry Pi. Berdasarkan hasil perhitungan, *delay* yang didapat dari Raspberry Pi 2 adalah 3,4 detik, sedangkan *delay* yang didapat dari Raspberry Pi B+ adalah 3,6 detik. Dengan kata lain, dapat disimpulkan bahwa proses pengiriman suara tidak banyak mempengaruhi performa Raspberry Pi.

Pengujian interferensi pengiriman suara dilakukan dengan parameter jarak dan interferensi jaringan. Pengujian dilakukan dengan cara memberikan jarak antar Raspberry Pi dan *router*.

Ketika pengujian selesai dilakukan, dapat disimpulkan bahwa kualitas sinyal sangat mempengaruhi *delay* pengiriman suara. Apabila Raspberry Pi diletakkan jauh dari *router*, maka *delay* dapat bertambah hingga 5 detik.

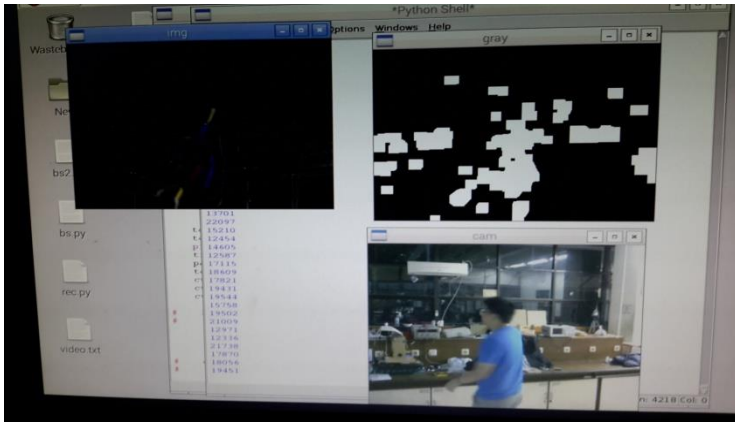
Pengujian interferensi dilakukan dengan cara menjalankan sistem di berbagai tempat sesuai dengan banyaknya interferensi, salah satunya adalah banyaknya *router*. Hal ini disebabkan karena pengiriman gambar dilakukan dengan protokol SSH, yaitu protokol yang biasa digunakan untuk mengakses komputer dari jarak jauh.

Ketika pengujian dilakukan pada tempat yang tidak memiliki interferensi, pengiriman suara dapat dilakukan dengan lancar dan tidak ada gangguan. Ketika pengujian dilakukan di tempat yang memiliki banyak jaringan, pengiriman suara akan terganggu dan seringkali tidak berhasil.

Dari pengujian interferensi, dapat disimpulkan bahwa pengiriman suara sangat bergantung pada kualitas jaringan dan kondisi interferensi ruangan. Apabila dijalankan pada ruangan yang memiliki banyak interferensi, maka pengiriman akan terganggu.

4.3 Pengujian Pendeteksi Gerakan

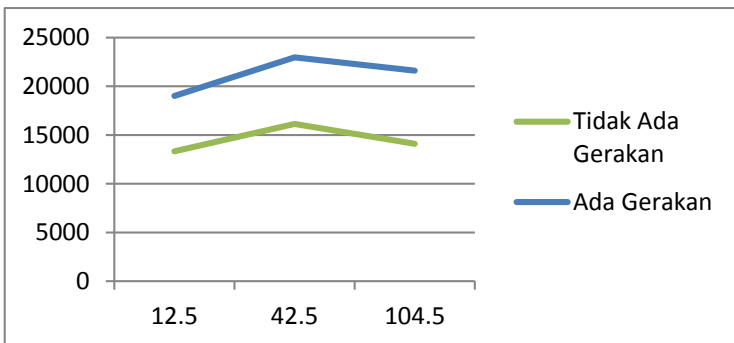
Pengujian ini bertujuan untuk melihat apakah program berjalan dan dapat mendeteksi gerakan. Pengujian juga dilakukan untuk melihat hubungan antara *threshold* dan intensitas cahaya. Pengujian dilakukan dengan cara membandingkan bit putih yang terhitung ketika terjadi gerakan atau tidak.



Gambar 4.5 Pengujian Pendeteksi Gerakan

Pengujian dilakukan di Lab Elektronika Terpadu ruangan B202 Teknik Elektro FTI ITS. Bit putih yang dihitung tertampil pada *window gray*, sedangkan angka tersebut menunjukkan banyaknya bit. Ketika tidak terdapat gerakan, bit putih berjumlah antara 14000 hingga 15000. Ketika terdapat gerakan, bit putih berjumlah antara 19000 hingga 22000.

Pengujian hubungan antara intensitas cahaya dan jumlah bit putih dilakukan dengan menggunakan Lux Meter. Pengujian dilakukan dengan tiga kondisi, yaitu lampu dihidupkan sepenuhnya, lampu dihidupkan sebagian, dan lampu dihidupkan sedikit. Nilai bit putih diambil sebanyak 10 data dan dirata-rata.



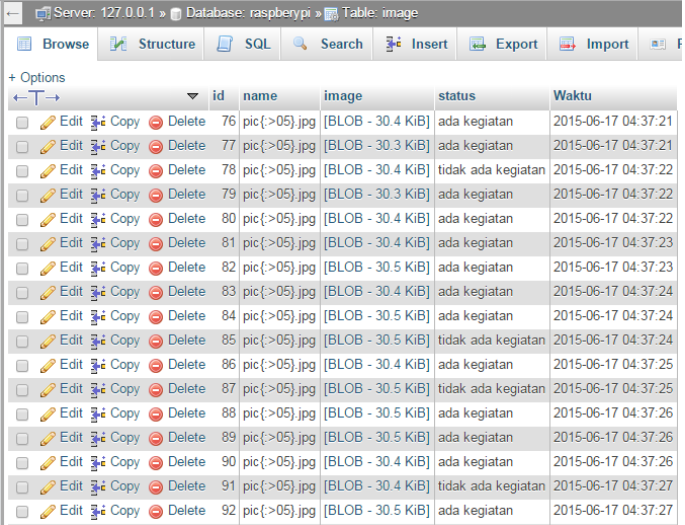
Gambar 4.6 Grafik Hubungan Intensitas Cahaya dan Bit Putih

Ketika pengujian dilakukan, intensitas cahaya dimana bit putih terdeteksi paling banyak adalah 42,5 lux. Jika intensitas cahaya semakin tinggi atau semakin rendah maka bit putih akan semakin sedikit terdeteksi. Untuk mendeteksi gerakan, *threshold* harus diatur diantara kondisi tidak ada gerakan dan ada gerakan.

4.4 Pengujian Sistem

Pengujian ini bertujuan untuk melihat apakah setiap komponen dapat saling berintegrasi untuk memenuhi tujuan sistem. Pengujian ini terdiri dari pengujian pengiriman data dilanjutkan dengan pengujian penyimpanan data di *database*, pengujian pengeluaran gambar, pengujian tampilan, dan pengujian monitoring gerakan.

Pengujian dilakukan dengan cara menjalankan semua sistem secara bersamaan dan melihat apakah sistem dapat bekerja dengan baik. Pengujian dimulai dari menjalankan program pengiriman gambar dan monitoring di Raspberry Pi, menjalankan program penyimpanan gambar di *server*, dan menjalankan program penampilan gambar di *server*.



	id	name	image	status	Waktu
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	76	pic{>05}.jpg	[BLOB - 30.4 KiB]	ada kegiatan	2015-06-17 04:37:21
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	77	pic{>05}.jpg	[BLOB - 30.3 KiB]	ada kegiatan	2015-06-17 04:37:21
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	78	pic{>05}.jpg	[BLOB - 30.4 KiB]	tidak ada kegiatan	2015-06-17 04:37:22
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	79	pic{>05}.jpg	[BLOB - 30.3 KiB]	ada kegiatan	2015-06-17 04:37:22
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	80	pic{>05}.jpg	[BLOB - 30.4 KiB]	ada kegiatan	2015-06-17 04:37:22
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	81	pic{>05}.jpg	[BLOB - 30.4 KiB]	ada kegiatan	2015-06-17 04:37:23
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	82	pic{>05}.jpg	[BLOB - 30.5 KiB]	ada kegiatan	2015-06-17 04:37:23
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	83	pic{>05}.jpg	[BLOB - 30.4 KiB]	ada kegiatan	2015-06-17 04:37:24
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	84	pic{>05}.jpg	[BLOB - 30.5 KiB]	ada kegiatan	2015-06-17 04:37:24
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	85	pic{>05}.jpg	[BLOB - 30.5 KiB]	tidak ada kegiatan	2015-06-17 04:37:24
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	86	pic{>05}.jpg	[BLOB - 30.4 KiB]	ada kegiatan	2015-06-17 04:37:25
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	87	pic{>05}.jpg	[BLOB - 30.5 KiB]	tidak ada kegiatan	2015-06-17 04:37:25
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	88	pic{>05}.jpg	[BLOB - 30.5 KiB]	ada kegiatan	2015-06-17 04:37:26
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	89	pic{>05}.jpg	[BLOB - 30.5 KiB]	ada kegiatan	2015-06-17 04:37:26
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	90	pic{>05}.jpg	[BLOB - 30.4 KiB]	ada kegiatan	2015-06-17 04:37:26
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	91	pic{>05}.jpg	[BLOB - 30.4 KiB]	tidak ada kegiatan	2015-06-17 04:37:27
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	92	pic{>05}.jpg	[BLOB - 30.5 KiB]	ada kegiatan	2015-06-17 04:37:27

Gambar 4.7 Database

Gambar 4.7 menunjukkan database sistem saat sedang berjalan. Data yang tersimpan di database adalah nomor gambar, gambar yang dapat disimpan, kondisi gerakan, dan waktu penyimpanan gambar. Waktu penyimpanan didapat dari waktu *server*.



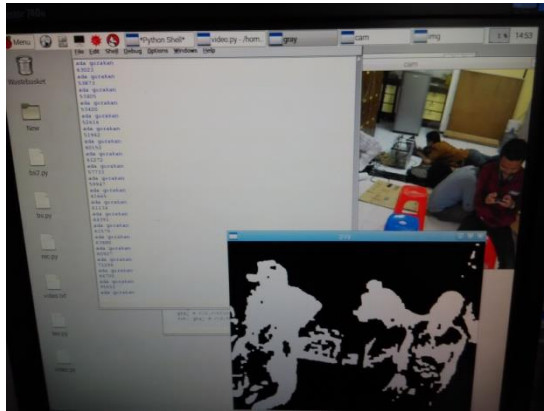
Gambar 4.8 Sistem Berjalan

Gambar 4.8 menunjukan *webpage* ketika sistem sedang berjalan. Gambar kiri menunjukkan tampilan kamera 1 yang tersambung dengan Raspberry Pi 2, sedangkan gambar kanan menunjukkan tampilan kamera 2 yang tersambung dengan Raspberry Pi B+. Status atas menunjukkan status kamera 1, sedangkan status bawah menunjukkan status kamera 2.

Dari percobaan diatas, dapat disimpulkan bahwa pengiriman data berhasil di kedua Raspberry Pi, akan tetapi transmisi data Raspberry Pi 2 lebih cepat jika dibandingkan dengan Raspberry Pi B+, sehingga apabila *delay* pengiriman disesuaikan dengan Raspberry Pi 2, transmisi data Raspberry Pi B+ tidak akan dapat mengikuti sehingga pada akhirnya kamera 2 akan menunjukkan gambar kosong.

Setelah terbukti bahwa sistem dapat bekerja satu sama lain, sistem diuji apakah dapat digunakan di dalam kelas. Pengujian dilakukan dengan cara mensimulasikan kondisi ujian. Simulasi berlangsung pada Lab Elektronika Terpadu ruangan B202 Jurusan Teknik Elektro FTI ITS.

Kondisi ujian yang disimulasikan adalah kondisi ujian tertulis tanpa membuka catatan (ujian tertutup), sehingga gerakan yang terlalu banyak akan langsung menimbulkan kecurigaan. Gerakan yang diuji adalah gerakan melihat catatan/peralatan elektronik dan gerakan melihat pekerjaan teman.



Gambar 4.9 Simulasi Ujian

Gambar 4.9 menunjukkan simulasi yang telah dilakukan. apabila gerakan yang dilakukan terlalu banyak maka jumlah bit putih akan melebihi *threshold*, yaitu 4000. Gerakan yang dinilai terlalu banyak adalah gerakan yang melebihi 25% gerakan tubuh, yaitu gerakan 1 lengan. Apabila gerakan yang terjadi hanya gerakan menulis, maka gerakan tersebut tidak akan melebihi *threshold*.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Sistem monitoring yang telah dibuat dapat memonitor dua ruangan yang berbeda secara langsung dengan batasan wilayah WLAN yang tersedia. Pengiriman data dilakukan melalui WLAN dan tersimpan di *database*. *Server* akan menyimpan gambar dan menampilkannya. *Frame per second* kecil, dikarenakan performa Raspberry Pi yang terbatas. Hal ini membuat gerakan tidak akan terdeteksi apabila terlalu cepat.

Pengiriman suara hanya dapat berhasil di tempat yang tidak memiliki interferensi. Hal ini disebabkan pengiriman data menggunakan protokol SSH dipengaruhi oleh keberadaan interferensi.

Pendeteksi gerakan aktif apabila terjadi gerakan terlalu banyak dan melebihi *threshold*. Jarak efektif pendeteksi adalah 2 hingga 10 meter. Presentase gerakan yang terdeteksi bergantung pada *threshold*, jarak, dan banyak objek. Apabila digunakan untuk mendeteksi 1 objek dengan jarak 5 meter dan *threshold* 4000, maka sistem akan menangkap gerakan apabila objek bergerak sebanyak 25%.

5.2 Saran

Berdasarkan pengujian yang telah dilakukan penulis, pengembangan sistem lebih lanjut adalah:

1. Penggunaan lensa *fish eye* agar jangkauan lebih luas.
2. Penggunaan komputer yang memiliki performa lebih bagus dari Raspberry Pi.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

- [1]. Simon Monk, "*Raspberry Pi Cookbook*", O'Reilly Media, Desember 2013.
- [2]. _____, "*Quick Start Guide*", 2012, <URL: <https://www.raspberrypi.org/wp-content/uploads/2012/12/quick-start-guide-v1.1.pdf>>.
- [3]. _____, "*MySQL and PHP Manual*", Oracle, 2015.
- [4]. Tom Harris, "*How Speakers Work*", < URL: <http://electronics.howstuffworks.com/speaker.htm>
- [5]. _____, "*Webcam*", <URL: <http://en.wikipedia.org/wiki/Webcam>>.
- [6]. John Morris., "How to Insert Form Data Into a MySQL Database Using PHP", April 2015 <URL: <http://www.johnmorrisonline.com/how-to-insert-form-data-into-a-mysql-database-using-php/>>.
- [7]. Andrew Hague, Graham Hastings, Michael Kolling, Andrew Hague, Ben Croston, Adrian Oldknow, Brian Lockwood, Clive Beale, "*The Raspberry Pi Education Manual, Version 1.0*", 2012.
- [8]. _____, "*PulseAudio-the next generation sound system*", Mandriva, 2008.
- [9]. _____, "*Introduction*", <URL: <http://docs.opencv.org/modules/core/doc/intro.html#introduction> >.
- [10]. _____, "*OpenCV*", <URL: <https://en.wikipedia.org/wiki/OpenCV>>.
- [11]. _____, "*Internet Protocol Suite*", <URL: http://en.wikipedia.org/wiki/Internet_protocol_suite>.
- [12]. _____, "*Database Definition*", <URL: <http://searchsqlserver.techtarget.com/definition/database>>.
- [13]. _____, "*Router*", <URL: <http://www.pcmag.com/encyclopedia/term/50637/router>>.
- [14]. _____, "*WirelessLAN*", <URL: https://en.wikipedia.org/wiki/Wireless_LAN>.
- [15]. _____, "*Internet Protocol Suite*", <URL: https://en.wikipedia.org/wiki/Internet_protocol_suite>.
- [16]. _____, "*Python*", <URL: <https://www.python.org/about/>>.
- [17]. TP-Link, "*150 Mbps Wireless N Nano USB Adapter TL-WN725N*", < URL: <http://www.tp-link.com>

- link.com/lk/products/details/?model=TL-WN725N#spec>.
- [18]. TP-Link, “*3G/4G Wireless N Router TL-MR3220*”, < URL:
http://www.tp-link.com/en/products/details/cat-4691_TL-MR3220.html#specifications>.

LAMPIRAN

Script program Raspberry Pi:

```
import cv2
import numpy as np
import time
import MySQLdb

capture = cv2.VideoCapture(0)
capture.set(3,360)
capture.set(4,480)
db = MySQLdb.connect(host="192.168.0.100", user="raspi", passwd =
"raspi", db="rasberrypi")

cur = db.cursor()

i = 0

while True:

    i += 1

    ret, gambar = capture.read()
    ret,tes = capture.read()
    ret,back = capture.read()
    cv2.imwrite('/home/pi/Desktop/New/pic.jpg'.format(i), tes)

def read_image():
    fin = open("/home/pi/Desktop/New/pic.jpg")
    img = fin.read()
    return img
with db:
    data = read_image()
```

```

        cur.execute("INSERT INTO image(name,image)
VALUES('pic{:>05}.jpg',%s)", (data,))

```

```

frame = cv2.blur(tes, (5,5))
image = cv2.absdiff(back, frame)
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
ret, gray = cv2.threshold(gray, 10, 255, cv2.THRESH_BINARY)

```

```

kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (5,5))
gray = cv2.morphologyEx(gray, cv2.MORPH_CLOSE, kernel)
kernel = np.ones((5,5),np.uint8)
gray = cv2.erode(gray,kernel,iterations = 2)
gray = cv2.dilate(gray,kernel,iterations = 2)
if 0xFF & cv2.waitKey(200) == ord('q'):
    break

```

```

test = cv2.countNonZero(gray)
test = int(test)
print(test)
treshhold = int(7500)
pause = 0
text = 0
cv2.imshow('gray',gray)
cv2.imshow('cam',back)
cv2.imshow('img',image)

```

```

if test < threshhold:
    print 'tidak ada orang'
    with db:
        data = read_image()
        cur.execute("INSERT INTO image(name,image,status)
VALUES('pic{:>05}.jpg',%s,'tidak ada kegiatan')", (data,))
    elif test >= threshhold :

```

```

    print 'ada orang'
    with db:
        data = read_image()
        cur.execute("INSERT INTO image(name,image,status)
VALUES('pic{:>05}.jpg',%s,'ada kegiatan')", (data,))

cv2.destroyAllWindows()

```

Script penyimpanan data:

```

<html>
<body>
<?php
ini_set('max_execution_time', 200);
$con=mysql_connect("127.0.0.1","root","");
$db=mysql_select_db("raspberrypi",$con);
$id = 1;
$is = 1;
while (true == true){
$qry="SELECT image FROM image WHERE id=$id";

$result=mysql_query($qry);
$fvp=fopen(($id)."png","w") or die("unable to open file");
$row = mysql_fetch_row($result);
fwrite($fp,$row[0]);
$id = $id+1;

$qry1="SELECT image FROM image2 WHERE id=$is";

$result1=mysql_query($qry1);
$fvp1=fopen("a".($is)."png","w") or die("unable to open file");
$row1 = mysql_fetch_row($result1);
fwrite($fp1,$row1[0]);
$is = $is+1;

```

```

usleep(500000);

}
fclose($fp);
fclose($fp1);
mysql_close($con);
?>
</body>
</html>

```

Script webpage:

```

<?php
$page = $_SERVER['PHP_SELF'];
$sec = "0.5";

?>
<html>
<head>
<meta http-equiv="refresh" content="<?php echo $sec?>;URL='<?php
echo $page?>'">
<title>CCTV AKTIVITAS KELAS X DAN KELAS Y</title>
</head>
<body>

<table width="730" border="1">
<tr>
<th colspan="3" scope="col"><div align="center">AKTIVITAS
KELAS</div></th>
</tr>
<tr>
<td width="365"><div align="center">

```

```

<?php
include 'id.php';
echo $id;
$id=$id+1;
$img= ($id).".png" ;
$image = '';
echo($image);
?>
</div></td>

<td height="200" colspan="2"><div align="center">
<?php
include 'is.php';
echo $is;
$is=$is+1;
$img1= "a".($is).".png" ;
$image1 = '';
echo($image1);
?>
</div></td>

</tr>
</table>
</body>
</html>

<?php
include 'id.php';
include 'is.php';
$con=mysql_connect("127.0.0.1","root","");
$db=mysql_select_db("raspberrypi",$con);
$qry="SELECT status FROM image WHERE id=$id";
$result=mysql_query($qry);
$row = mysql_fetch_array($result, MYSQL_ASSOC);
$fromid = $row['status'];
echo "$fromid.<br>";

```

```

$qry1="SELECT status FROM image2 WHERE id=$is";
$result1=mysql_query($qry1);
$row1 = mysql_fetch_array($result1, MYSQL_ASSOC);
$fromid1 = $row1['status'];
echo $fromid1;
$id=$id+1;
$var_str = var_export($id, true);
$var = "<?php\n\n$id = $var_str;\n\n?>";
file_put_contents('id.php',$var);
$is=$is+1;
$var_str1 = var_export($is, true);
$var1 = "<?php\n\n$is = $var_str1;\n\n?>";
file_put_contents('is.php',$var1);
?>

```


BIODATA PENULIS



Penulis bernama lengkap Andri Yasmin lahir pada tanggal 5 Oktober 1993, sebagai anak keempat dari empat bersaudara. Penulis mengawali kegiatan pendidikan formal di SD Frater Xaverius 2 Palembang, yang kemudian dilanjutkan di SMP Xaverius 1 Palembang, kemudian SMA Xaverius 1 Palembang, dan pada tahun 2011 penulis diterima sebagai mahasiswa di jurusan Teknik Elektro ITS. Saat kuliah, penulis turut berpartisipasi sebagai asisten praktikum di bidang studi elektronika serta aktif pula dalam Keluarga

Mahasiswa Katolik St. Ignatius ITS Surabaya dan Electrical Engineering Event 2014.

Email : andriyasmin0510@gmail.com